

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:

Robert Betros

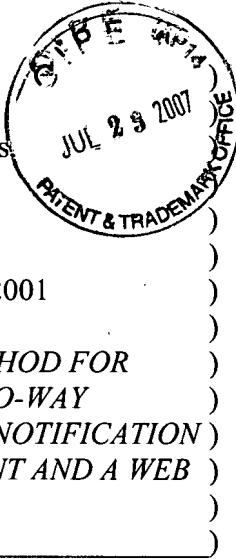
Serial No.: 09/766,439

Filed: January 19, 2001

For: *SYSTEM AND METHOD FOR
MAINTAINING TWO-WAY
ASYNCHRONOUS NOTIFICATION
BETWEEN A CLIENT AND A WEB
SERVER*

Group Art Unit: 2142

Examiner: Kevin Lin



San Diego, California
July 17, 2007

MAIL STOP AMENDMENT
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

EXPERT DECLARATION UNDER 37 C.F.R. 1.132

Dear Sir:

This is an Expert Declaration in Support of Traversal, submitted under 37 C.F.R.
§ 1.132.

Relationship to the Application

I, Brad Tipler, have been retained to provide an independent analysis and opinion as to the level of experimentation necessary to practice the invention described in the above identified application. For this work I am being compensated at a rate of \$300.00 per hour.

The Subject Invention

This invention relates to computer communications. More particularly this invention relates to a system and method for maintaining direct, two-way asynchronous communication between a client and a web server.

Basis of the Opinion

The technical basis of this declaration stems from my 25 years in computer communications in both industry and academia.

I have met with Dr. Steve Moore, attorney of record in the present application. Dr. Moore has provided me with the legal basis for Enablement under 35 U.S.C. 112 and specifically the factors for determining "undue experimentation" set forth by the court in *In Re Wands*.

Enablement Analysis

In view of the In Re Wands factors I conclude that the specification gives sufficient guidance to one of ordinary skill.

In Re Wands Factors: The factors discussed herein are (1) the breadth of the claims; (2) the nature of the invention; (3) the state of the prior art; (4) the level of one of ordinary skill; (5) the level of predictability in the art; (6) the amount of direction provided by the inventor; and (7) the quantity of experimentation needed to make or use the invention. Not discussed here is the additional factor of the existence of working examples. This factor is outside of the scope of this declaration.

1. Breadth of the Claims

The scope of the claims is commensurate with the specification and not overly broad. When considering the scope of independent claim 1 it is important to note that the scope does not encompass all particular implementations for performing two-way asynchronous communications between a client and a server. The claim is limited by communicating through a single socket connection, using the HTTP protocol, within a single HTTP transaction, it is further limited by a client device sending particular information from the server to the client and sending information from the client to the server. The particular information and the information being communicated in a protocol other than HTTP. Further, the server needs to be able to send the particular information to the client without receiving a request from the client.

Further independent claim 9 includes “means for” limitations, which as I understand, are limited to the disclosed embodiments and their equivalents.

Additionally, independent claim 20 requires that the client open the socket connection, the server to execute software that enables two-way asynchronous communication wholly within a single HTTP transaction. The two-way communication that is enabled is a client device sending particular information from the server to the client and sending information from the client to the server. The particular information and the information being communicated in a protocol other than HTTP.

Independent claim 21 includes "means-for" limitations that are limited to the disclosed embodiments and their equivalents.

It is therefore my opinion that, when considering the limitations on the breadth of these claims, they are adequately described in the specification.

2. The Nature of the Invention.

As stated above, this invention relates to computer communications. More particularly this invention relates to a system and method for maintaining direct, two-way asynchronous communication between a client and a web server. Computer communication protocols have been known by persons of ordinary skill for a significant time. For examples of computer communications protocols, see the foundational work of Drs. Fouad Tobagi and Leonard Kleinrock in their 4-part series of publications on the state of the art between December 1975 and October 1977. These references are attached as Exhibits A through D. Further the HTTP specification was well known at the time of the invention. A draft copy of the HTTP 1.0 specification, dated February 19, 1996 is attached as Exhibit E. The nature of the invention is novel and non-obvious advancement in a well known field.

3. The State of the Prior Art.

By January 19, 2001 those of ordinary skill in the art of computer communications protocols, and more specifically HTTP protocols, would be familiar with the concepts underlying Exhibits A through E. Internet programmers would need an in-depth working knowledge of at least HTTP 1.0 specification in order to be functional in the art. Internet programmers were at the core of the “dot com” era which is discussed in Wikipedia article attached as Exhibit F. The state of the art in HTTP programming, and the level of those in the art was high. Because of the plethora of activity in Internet based communications, the state of the art related to HTTP based communications was well developed by the date of the invention.

4. The Level of One of Ordinary Skill.

As stated above, one of ordinary skill would need to be familiar with Internet programming in general, HTTP protocols in specific. The state of the art was fully developed by the time of the invention. Further, since the entire “dot com” economy was dependent on ordinary skilled artisans. In further view of Exhibit F it is my opinion that the level of skill in computer communications and particularly in HTTP programming was high.

5. The Level of Predictability in the Art

The design of communications protocols and the subsequent computer programming for communications is extremely predictable. A computer will execute the code you provide. This is well known to all in the art.

6. The Amount of Direction Provided by the Inventor

The level of specificity provided in the application is specific enough to allow one of ordinary skill in the art to make and use the invention without undue experimentation. For example, the inventor disclosed an exemplary CGI as one implementation. CGI programming was very well known by 2001. For example, In section 1.1 of Shishig Gundavaram's "CGI Programming on the World Wide Web", Published by O'Reilly in March 1996 stated "CGI isn't magic, it's just programming with some special types of input and a few strict rules on program output." A copy of the relevant portions of this text are attached as Exhibit G. As discussed above, Internet programming skills were well known in the art, the level of predictability of programming is very high, and internet tools like CGI aren't magic, they are just programming.

7. The Quantity of Experimentation Needed to Make or Use the Invention

It is my opinion that the quantity of experimentation needed to make and use the invention is very low. Given the direction provided by the inventor in the claims it would be straightforward to produce a working system.

In summary, after reviewing the disclosure, I believe the disclosure enables one of ordinary skill to make and use the invention without undue experimentation.

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment or both, under Section 1001 of Title 18

of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

Respectfully submitted,

July 17, 2007

Date



Brad Tippler